Opinion mining of text documents written in Macedonian language

Andrej Gajduk* and Ljupco Kocarev*‡§

***Keywords*—opinion mining, classification, natural language processing, Macedonian language**

***Abstract*—The ability to extract public opinion from web portals such as review sites, social networks and blogs will enable companies and individuals to form a view, an attitude and make decisions without having to do lengthy and costly researches and surveys. In this paper machine learning techniques are used for determining the polarity of forum posts on kajgana which are written in Macedonian language. The posts are classified as being positive, negative or neutral. We test different feature metrics and classifiers and provide detailed evaluation of their participation in improving the overall performance on a manually generated dataset. By achieving 92% accuracy, we show that the performance of systems for automated opinion mining is comparable to a human evaluator, thus making it a viable option for text data analysis. Finally, we present a few statistics derived from the forum posts using the developed system.**

I. Introduction

The World Wide Web (Web) has tremendously influenced our lives by changing the way we manage and share the information. Today, we are not only static observers and receivers of information, but in turn, we actively change the information content and/or generate new pieces of information. In this way, the entire community becomes a writer, in addition to being a reader. Different mediums, such as blogs, wikis, forums and social networks, exist in which we can express ourselves by posting information and giving opinion on various subjects, ranging from politics and health to product reviews and travelling.

Sentiment analysis (also referred as opinion mining) concerns application of natural language processing, computational linguistics, and text analytics to identify and extract subjective information in source materials. Opinion Mining operates at the level of documents, that is, pieces of text of varying sizes and formats, e.g., web pages, blog posts, comments, or product reviews. We assume that each document discusses at least one topic, that is, a named entity, event, or abstract concept that is mentioned in a document. Sentiment is the authors attitude, opinion, or emotion expressed on a topic. Although sentiments are expressed in natural language, they can in some cases be translated to a numerical or other scale, which facilitates further processing and analysis. Since the palette of human emotions is so vast and it is hard to select even the basic ones, most of the authors in the NLP community work with representation of sentiments according to their polarity, which means positive or negative evaluation of the meaning of the sentiment.

It is now well-documented that the opinions/views expressed on the web can be influential to readers in forming their opinions on some topic [1], and therefore, they are an important factor taken into consideration by product vendors [2] and policy makers [3]. There exists evidence that this process has significant economic effects [4]–[6]. Moreover, the opinions aggregated at a large scale may reflect political preferences [7], [8] and even improve stock market prediction [9]. For the recent surveys on sentiment analysis or opinion mining we refer readers to [10]–[12].

The outline of the paper is as follows. In Section 2 the problem of opinion mining is formally defined. The proposed approach is outlined in Section 3. In Section 4 we give details about the datasets used in our experiments. In Section 5 the performance achieved using the different feature representation, classifiers and other text processing techniques is compared. A few statistics on the forum posts on kajgana derived using opinion mining are presented in Section 6. Section 7 concludes this paper.

II. Problem Definition

In our experiment, we accept the classification of opinions according to their polarity i.e. polarity classification, used by the majority of authors [2], [10]. Pang and Lee [10] define polarity as the point on the evaluation scale that corresponds to our *positive* or *negative* evaluation of the meaning of the expressed opinion. However, not all texts are opinionated, so the method proposed by [13]

* Macedonian Academy of Sciences and Arts.
‡ Ss. Cyril and Methodius University, Faculty of Computer Science and Engineering, Skopje.
§ BioCircuits Institute, University of California at San Diego.

which rates subjectivity and polarity separately is used. The problem is defined as follows:
Given a piece of text, decide whether it is subjective or objective, then assuming that the overall opinion in it is about one single issue or item, classify the opinion in subjective posts as falling under one of the two categories: positive or negative.

III. Proposed approach

A. Data representation

Text data in machine learning is commonly represented by using the bag-of-features method [14]–[17]. This method is described as follows: let $D = \{f_1, \ldots, f_m\}$ be a predefined set of m features that can appear in a forum post. We will refer to D as a feature dictionary. The features in the dictionary can be unigrams i.e. words such as *great* and *wasteful*, bigrams i.e. word pairs such as *not comfortable* or n-grams in the general case. Every post is represented by a vector of real numbers which correspond to a single feature in the feature dictionary. These values are computed using four different feature metrics.

- n-gram presence

$$presence_i^p = \begin{cases} 1, & \text{if } t_i^p \neq 0 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

- n-gram count

$$count_i^p = t_i^p \quad (2)$$

- n-gram frequency

$$freq_i^p = \frac{t_i^p}{\sum_j t_j^p} \quad (3)$$

- n-gram frequency-inverse document frequency

$$ifreq_i^p = freq_i^p \log \frac{||P||}{||P_i||} \quad (4)$$

In (1–4) t_i^p is the number of occurrences of the ith n-gram in the post p, P is a set of all the posts and P_i is a set of posts containing at least one occurrence of the ith n-gram. Unigrams are the most commonly used in text mining, although some authors [18] recommend using bigrams. Their arguments include dealing with word negation and emphasizing which are very important in the domain of polarity classification.

B. Classifiers

The proposed feature metrics are evaluated using the two classifiers preferred by the majority of researchers in text classification [19]–[22].

- Support Vector Machines
- Naive Bayes

As discussed earlier, the classification will take place in two phases. First, subjectivity classification is performed where the comment is rated as either subjective or objective. Then if the post is subjective, it is classified as being either positive or negative. The latter will be referred to as polarity classification.

C. Preprocessing

Stop Words: Filtering stop words is a common practice in text mining [23]–[26]. Stop words are words with no informational value, such as function and lexical words. A suitable list of stop words in Macedonian language is difficult to obtain so one had to be manually prepared for this experiment. The list of stop words constitutes of 170 entries.

Stemming: Stemming has been extensively used to increase the performance of information retrieval systems for many international languages such as: English, French, Portuguese, to name a few [27], [28]. Stemming is a technique which aims to reduce a word to its stem or root form. Thus, literally different words that share a common stem may be abstracted as a single informational entity. There are several common approaches to stemming as categorized in [29] namely affix removal method, successor variety method, n-gram method and table lookup method. Affix removal which includes algorithms such as Lovins or Porter, is the most popular method, but relies heavily on manually defined rule sets. A good rule set for Macedonian is yet to be defined, which is why we decided to use a stemming method that relies on nothing more but the set of words that need to be stemmed. This method is called peak-and-plateau and is based on tries. For a more detailed explanation to this method we refer readers to [30].

D. Rule bigrams

Some authors propose a different way of incorporating bigrams into the feature vector [31]–[34] which will be refereed to as *rule bigrams*. According to this approach all negatory words are appended a tag *e.g. not* to the word following the negatory word in the sentence. Thus

Accuracy	SVM	NB
Presence	0.76	0.64
Count	0.73	0.55
Frequency	0.72	0.61
IFrequency	0.94	0.78

TABLE I: Accuracy, no preprocessing

Accuracy	SVM	NB
Presence	0.76	0.63
Count	0.72	0.56
Frequency	0.70	0.60
IFrequency	0.74	0.62

TABLE II: Accuracy, with preprocessing

the bigram *not good* becomes the unigram *notgood*. This method is adopted and expanded to emphasizery words, thus transforming bigrams such as *most disgusting* and *very disgusting* into the same unigram *e.g. verydisgusting*. This approach is adequate when using unigram presence as a feature vector, but we propose an alteration when applying it in combination with other feature metrics that rely on counting the unigram occurrences. Any occurrence of an unigram preceded by an emphasizing word is counted as two occurrences of the corresponding unigram i.e. $\hat{t}_i^p = 2t_i^p$, whereas any occurrence of an unigram preceded by a negatory word is considered as -1 occurrence of the corresponding unigram i.e. $\hat{t}_i^p = -t_i^p$.

IV. Dataset

The domain used in this study is forum posts which are written in Macedonian language from the kajgana forum. Forum posts tend to be less focused and organized than other text documents such as product reviews for instance, and consist predominantly of informal text. The posts on kajgana are grouped into 47 disjoint topics which are then divided into subtopics (over 50,000) and are 60 words long on average. There are a total of 4 million unique words in the posts. In our experiment, we ignored words that have less than 5 occurrences in order to reduce the total dictionary size and to eliminate type errors. This left us with 800,000 unique words. A total of 800 posts were manually tagged of which 260 are positive, 260 are negative and 280 are objective posts. This dataset will be used for evaluations on the different classifiers and feature representations. All evaluations are done using 10-fold cross validation to avoid over-fitting.

V. Results

First, the aforementioned feature representations using unigrams in combination with the two proposed classifiers are evaluated. Inverse frequency the best feature representation followed by presence (Table. I). As for classifiers, SVM outperforms NB on every feature representation.

Surprisingly, stemming and stop words removal reduces accuracy (Table II). More specifically the accuracy drops from 0.94 to 0.74 when using an SVM classifier and from 0.78 to 0.62 when using an NB classifier. One possible reason is that the word stemming algorithms does not perform well for the Macedonian language.

Presence	SVM	NB
Unigrams only	0.76	0.63
Bigrams only	0.54	0.52
Unigrams bigrams	0.79	0.67

IFrequency	SVM	NB
Unigrams only	0.74	0.62
Bigrams only	0.55	0.52
Unigrams bigrams	0.75	0.62

TABLE III: Accuracy, bigrams

As mentioned earlier the proposed feature representations can be applied to n-grams of any size, although so far only unigrams have been used. Next, we evaluate presence and ifrequency using bigrams, alone and in combination with unigrams (Table III). Bigrams alone are not good features, but when used in conjunction with unigrams they show a slight improvement when presence as feature representation is used from 0.76 to 0.78 with SVM and from 0.63 to 0.67 with NB.

Finally, in Table IV the accuracy when using rule bigrams (only negation rules, only emphasis rules and both together) are given. The results show that rule bigrams do not impact classification accuracy, with the exception of negation rules that achieves a slight increase in accuracy for unigram presence .

VI. Statistics

Using the combination of unigram ifrequency for a feature representation and SVM as a classifier some interesting properties of forum posts in general can be demonstrated. As stated above the forum posts are divided into several topics. Let us denote with p_t the number of positive posts and with n_t the number of negative posts for each topic t. The overall mood on the topic m_t is defined as

$$m_t = \frac{p_t}{p_t + n_t} \tag{5}$$

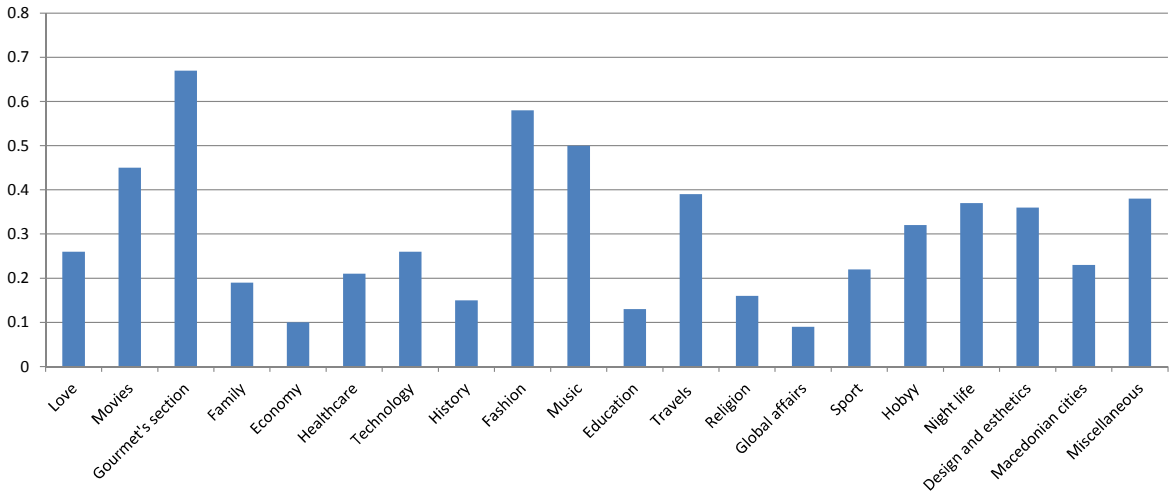

Fig. 1: Mood by topic

Presence	SVM	NB
Unigram	0.76	0.63
Negations only	0.78	0.62
Emphasizers only	0.76	0.61
Both	0.77	0.62

IFrequency	SVM	NB
Unigram	0.74	0.62
Negations only	0.73	0.59
Emphasizers only	0.74	0.59
Both	0.74	0.61

TABLE IV: Accuracy, rule bigrams

Interestingly, people are most positive when discussing food (Gourmets section) and fashion, but are extremely negative on global affairs and the economy (Fig. 1). .

In a similar fashion the posts can be grouped and their mood calculated by month as displayed in Fig. 2. The public mood is highest in spring (May and April), probably due to the good weather during these two months.

VII. Conclusion

In this paper forum posts written in Macedonian language are labeled as being positive, negative or objective. We show that this can be done with great accuracy using simple text feature extraction metrics such as unigram presence and standard classifiers such as Naive Bayes. The best accuracy is achieved by using a combination of unigram frequency-inverse document frequency for a feature metrics and support vector machines as a classifier: 0.96 on subjectivity classification, 0.96 on polarity classification or a total classification accuracy of 0.92. Additionally, we tested various techniques for improving the performance. Of these, word stemming and stop words removal had a negative effect on classification accuracy. The use of bigrams does not help with the classification task while using rule bigrams increases the accuracy only slightly in polarity classification.

References

[1] Y. Lin, J. Zhang, X. Wang, and A. Zhou, "Sentiment classification via integrating multiple feature presentations," in *Proceedings of the 21st international conference companion on World Wide Web*, pp. 569–570, ACM, 2012.

[2] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pp. 79–86, Association for Computational Linguistics, 2002.

[3] J. A. Horrigan, "Online shopping," *Pew Internet & American Life Project Report*, vol. 36, 2008.

[4] W. Antweiler and M. Z. Frank, "Is all that talk just noise? the information content of internet stock message boards," *The Journal of Finance*, vol. 59, no. 3, pp. 1259–1294, 2004.

[5] N. Archak, A. Ghose, and P. G. Ipeirotis, "Show me the money!: deriving the pricing power of product features by mining consumer reviews," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 56–65, ACM, 2007.

[6] J. A. Chevalier and D. Mayzlin, "The effect of word of mouth on sales: Online book reviews," tech. rep., National Bureau of Economic Research, 2003.

[7] T. Mullen and R. Malouf, "A preliminary investigation into sentiment analysis of informal political discourse.," in *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, pp. 159–162, 2006.

[8] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, "Predicting elections with twitter: What 140 characters reveal about political sentiment.," *ICWSM*, vol. 10, pp. 178–185, 2010.

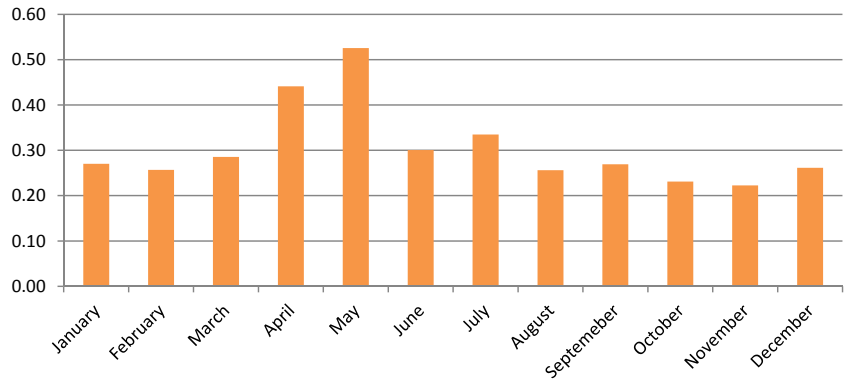

Fig. 2: Mood by month

[9] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, no. 1, pp. 1–8, 2011.

[10] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and trends in information retrieval*, vol. 2, no. 1-2, pp. 1–135, 2008.

[11] H. Tang, S. Tan, and X. Cheng, "A survey on sentiment detection of reviews," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10760–10773, 2009.

[12] M. Tsytsarau and T. Palpanas, "Survey on mining subjective data on the web," *Data Mining and Knowledge Discovery*, vol. 24, no. 3, pp. 478–514, 2012.

[13] N. Godbole, M. Srinivasaiah, and S. Skiena, "Large-scale sentiment analysis for news and blogs.," *ICWSM*, vol. 7, 2007.

[14] T. Nakagawa, K. Inui, and S. Kurohashi, "Dependency tree-based sentiment classification using crfs with hidden variables," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 786–794, Association for Computational Linguistics, 2010.

[15] J. Read, "Using emoticons to reduce dependency in machine learning techniques for sentiment classification," in *Proceedings of the ACL Student Research Workshop*, pp. 43–48, Association for Computational Linguistics, 2005.

[16] M. J. Paul, C. Zhai, and R. Girju, "Summarizing contrastive viewpoints in opinionated text," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pp. 66–76, Association for Computational Linguistics, 2010.

[17] M. Rushdi Saleh, M. T. Martín-Valdivia, A. Montejo-Ráez, and L. Ureña-López, "Experiments with svm to classify opinions in different domains," *Expert Systems with Applications*, vol. 38, no. 12, pp. 14799–14804, 2011.

[18] M. Zeng, Y. Yang, and W. Liu, "An approach of text sentiment analysis for public opinion monitoring system," in *Semantic Web and Web Science*, pp. 131–141, Springer, 2013.

[19] T. Mullen and N. Collier, "Sentiment analysis using support vector machines with diverse information sources.," in *EMNLP*, vol. 4, pp. 412–418, 2004.

[20] Q. Ye, B. Lin, and Y.-J. Li, "Sentiment classification for chinese reviews: A comparison between svm and semantic approaches," in *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, vol. 4, pp. 2341–2346, IEEE, 2005.

[21] M. Gamon, "Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis," in *Proceedings of the 20th international conference on Computational Linguistics*, p. 841, Association for Computational Linguistics, 2004.

[22] M. Koppel and J. Schler, "The importance of neutral examples for learning sentiment," *Computational Intelligence*, vol. 22, no. 2, pp. 100–109, 2006.

[23] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language," *Language resources and evaluation*, vol. 39, no. 2-3, pp. 165–210, 2005.

[24] P. Chaovalit and L. Zhou, "Movie review mining: A comparison between supervised and unsupervised classification approaches," in *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on*, pp. 112c–112c, IEEE, 2005.

[25] C. Strapparava and R. Mihalcea, "Semeval-2007 task 14: Affective text," in *Proceedings of the 4th International Workshop on Semantic Evaluations*, pp. 70–74, Association for Computational Linguistics, 2007.

[26] A. Esuli and F. Sebastiani, "Determining term subjectivity and term orientation for opinion mining.," in *EACL*, vol. 6, p. 2006, 2006.

[27] W. B. Frakes and C. J. Fox, "Strength and similarity of affix removal stemming algorithms," in *ACM SIGIR Forum*, vol. 37, pp. 26–30, ACM, 2003.

[28] J. Savoy, "Searching strategies for the hungarian language," *Information processing & management*, vol. 44, no. 1, pp. 310–324, 2008.

[29] D. Sharma, "Stemming algorithms: A comparative study and their analysis," *International Journal of Applied Information Systems*, vol. 4, no. 3, pp. 7–12, 2012.

[30] M. A. Hafer and S. F. Weiss, "Word segmentation by letter successor varieties," *Information storage and retrieval*, vol. 10, no. 11, pp. 371–385, 1974.

[31] M. Ghiassi, J. Skinner, and D. Zimbra, "Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network," *Expert Systems with Applications*, 2013.

[32] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic

compositionality over a sentiment treebank," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2013.

[33] S. Wang and C. D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pp. 90–94, Association for Computational Linguistics, 2012.

[34] H. Kang, S. J. Yoo, and D. Han, "Senti-lexicon and improved naïve bayes algorithms for sentiment analysis of restaurant reviews," *Expert Systems with Applications*, vol. 39, no. 5, pp. 6000–6010, 2012.